

IN THE CLAIMS

1. (previously presented) A method of detecting an intrusion in a communications network, the method comprising the steps of:

a) accessing, by a network intrusion detection process of a target computer system, communication to an application receive queue (ARQ) for an application running in an application layer of the target computer system, wherein the ARQ functions intermediate the application layer and a transport layer of a network protocol associated with said communications network to receive data packets for the application from the transport layer;

b) scanning for the application by the network intrusion detection process only the data packets accessed by the network intrusion detection process in a), wherein the data packets are directed to the application from a remote host via the communications network, and wherein the scanning is after the data packets have been processed by the transport layer and after the transport layer has passed the processed data packets for receipt by the application's ARQ;

c) determining if said scanned data packets are malicious; and

d) taking at least one action to prevent the application from processing data packets from the remote host to the application responsive to c) determining that any of the scanned data packets are malicious.

2. (previously presented) The method according to claim 1, wherein said at least one action includes terminating the application.

3. (original) The method according to claim 1, further comprising the step of transmitting to said application layer any data packets determined not to be malicious.

4. (original) The method according to claim 1, wherein said scanning and determining steps are implemented using a scan module.

5-6. (canceled)

7. (previously presented) The method according to claim 1, further comprising the step of obtaining data from said at least one ARQ.

8. (canceled)

9. (original) The method according to claim 1, further comprising the step of dispatching said data packets to one or more handlers for scanning, if said protocol is monitored.

10. (original) The method according to claim 1, wherein said scanning and determining steps are implemented using a scan daemon.

11. (previously presented) The method according to claim 1, further comprising the step of the target computer system generating fake, network-accessible services.

12. *(withdrawn) A method of preventing an intrusion in a communications network, the method comprising the steps of:*

disabling a network interface of a host if an idle time expires;

determining if any packets are to be transmitted; and

enabling said network interface if at least one packet is determined to be available to be transmitted.

13. (previously presented) A target computer system for detecting an intrusion originating from a remote host and communicated to the target computer system via a communications network, the target computer system comprising:

a storage unit for storing data and instructions for a processing unit; and
a processing unit coupled to said storage unit, said processing unit being programmed to perform steps responsive to the instructions, wherein the steps comprise:

a) accessing, by a network intrusion detection process of the target computer system, communication to an application receive queue (ARQ) for an application running in an application layer of the target computer system, wherein the ARQ functions intermediate the application layer and a transport layer of a network protocol associated with said communications network to receive data packets for the application from the transport layer;

b) scanning for the application by the network intrusion detection process only the data packets accessed by the network intrusion detection process in a), wherein the data packets are directed to the application from the remote host via the communications network, and wherein the scanning is after the data packets have been processed by the transport layer and after the transport layer has passed the processed data packets for receipt by the application's ARQ;

c) determining if said scanned data packets are malicious; and

d) taking at least one action to prevent the application from processing the data packets from the remote host to the application responsive to c) determining that any of the scanned data packets are malicious.

14. (previously presented) The system according to claim 13, wherein said at least one action includes terminating the application.

15. (original) The system according to claim 13, wherein said processing unit is programmed to transmit to said application layer any data packets determined not to be malicious.

16. (original) The system according to claim 13, wherein said processing unit is programmed to implement a scan module.

17-18. (canceled)

19. (previously presented) The system according to claim 13, wherein said processing unit is programmed to obtain data from said at least one ARQ.

20. (previously presented) The system according to claim 19, wherein said scanning is performed on data packets from said at least one ARQ.

21. (original) The system according to claim 13, wherein said processing unit is programmed to dispatch said data packets to one or more handlers for scanning, if said protocol is monitored.

22. (original) The system according to claim 13, wherein said scanning and determining are implemented using a scan daemon.

23. (previously presented) The system according to claim 13, wherein said processing unit is programmed to generate fake, network-accessible services.

24. *(withdrawn) A system of preventing an intrusion in a communications network, the system comprising:*

*a storage unit for storing data and instructions for a processing unit; and
a processing unit coupled to said storage unit, said processing unit being
programmed to disable a network interface of a host if an idle time expires, to
determine if any packets are to be transmitted, and to enable said network interface if
at least one packet is determined to be available to be transmitted.*

25. (previously presented) A computer program product stored on a computer-readable storage medium, the computer program product having instructions for execution by a computer, wherein the instructions, when executed by the computer, cause the computer to implement a method comprising the steps of:

a) accessing, by a network intrusion detection process of a target computer system, communication to an application receive queue (ARQ) for an application running in an application layer of the target computer system, wherein the ARQ functions intermediate the application layer and a transport layer of a network protocol associated with said communications network to receive data packets for the application from the transport layer;

b) scanning for the application by the network intrusion detection process only the data packets accessed by the network intrusion detection process in a), wherein the data packets are directed to the application from a remote host via the communications network, and wherein the scanning is after the data packets have been processed by the transport layer and after the transport layer has passed the processed data packets for receipt by the application's ARQ;

c) determining if said scanned data packets are malicious; and

d) taking at least one action to prevent the application from processing data packets from the remote host to the application responsive to c) determining that any of the scanned data packets are malicious.

26. (previously presented) The computer program product according to claim 25, wherein said at least one action includes terminating the application.

27. (previously presented) The computer program product according to claim 25, the steps further comprising transmitting to said application layer any data packets determined not to be malicious.

28. (previously presented) The computer program product according to claim 25, wherein said scanning and determining are implemented using a scan module.

29-30. (canceled)

31. (previously presented) The computer program product according to claim 25, the steps further comprising obtaining data from said at least one ARQ.

32. (canceled)

33. (previously presented) The computer program product according to claim 25, the steps further comprising dispatching said data packets to one or more handlers for scanning, if said protocol is monitored.

34. (previously presented) The computer program product according to claim 25, wherein said scanning and determining are implemented using a scan daemon.

35. *(withdrawn) A computer-readable medium of preventing an intrusion in a communications network, the computer-readable medium comprising:*
programmed instructions for disabling a network interface of a host if an idle time expires;
programmed instructions for determining if any packets are to be transmitted;
and
programmed instructions for enabling said network interface if at least one packet is determined to be available to be transmitted.

36. (previously presented) The method according to claim 1, wherein said at least one action includes modifying firewall rules to prevent reception of data packets from the host computer system.

37. (previously presented) The method according to claim 1, wherein the directing of the data packets to the application from the remote host is via a connection with the remote host on the communications network, and wherein said at least one action includes intimating the transport layer to tear down the remote host connection.

38. (previously presented) The method according to claim 37, wherein after intimating the transport layer to tear down the remote host connection, the target computer services requests on connections other than that remote host connection.

39. (previously presented) The system according to claim 13, wherein said at least one action includes modifying firewall rules to prevent reception of data packets from the host computer system.

40. (previously presented) The system according to claim 13, wherein the directing of the data packets to the application from the remote host is via a connection with the remote host on the communications network, and wherein said at least one action includes intimating the transport layer to tear down the remote host connection.

41. (previously presented) The system according to claim 40, wherein after intimating the transport layer to tear down the remote host connection, the target computer services requests on connections other than that remote host connection.

42. (previously presented) The computer program product according to claim 25, wherein said at least one action includes modifying firewall rules to prevent reception of data packets from the host computer system.

43. (previously presented) The computer program product according to claim 25, wherein the directing of the data packets to the application from the remote host is via a connection with the remote host on the communications network, and wherein said at least one action includes intimating the transport layer to tear down the remote host connection.

44. (previously presented) The computer program product according to claim 43, wherein after intimating the transport layer to tear down the remote host connection, the target computer services requests on connections other than that remote host connection.